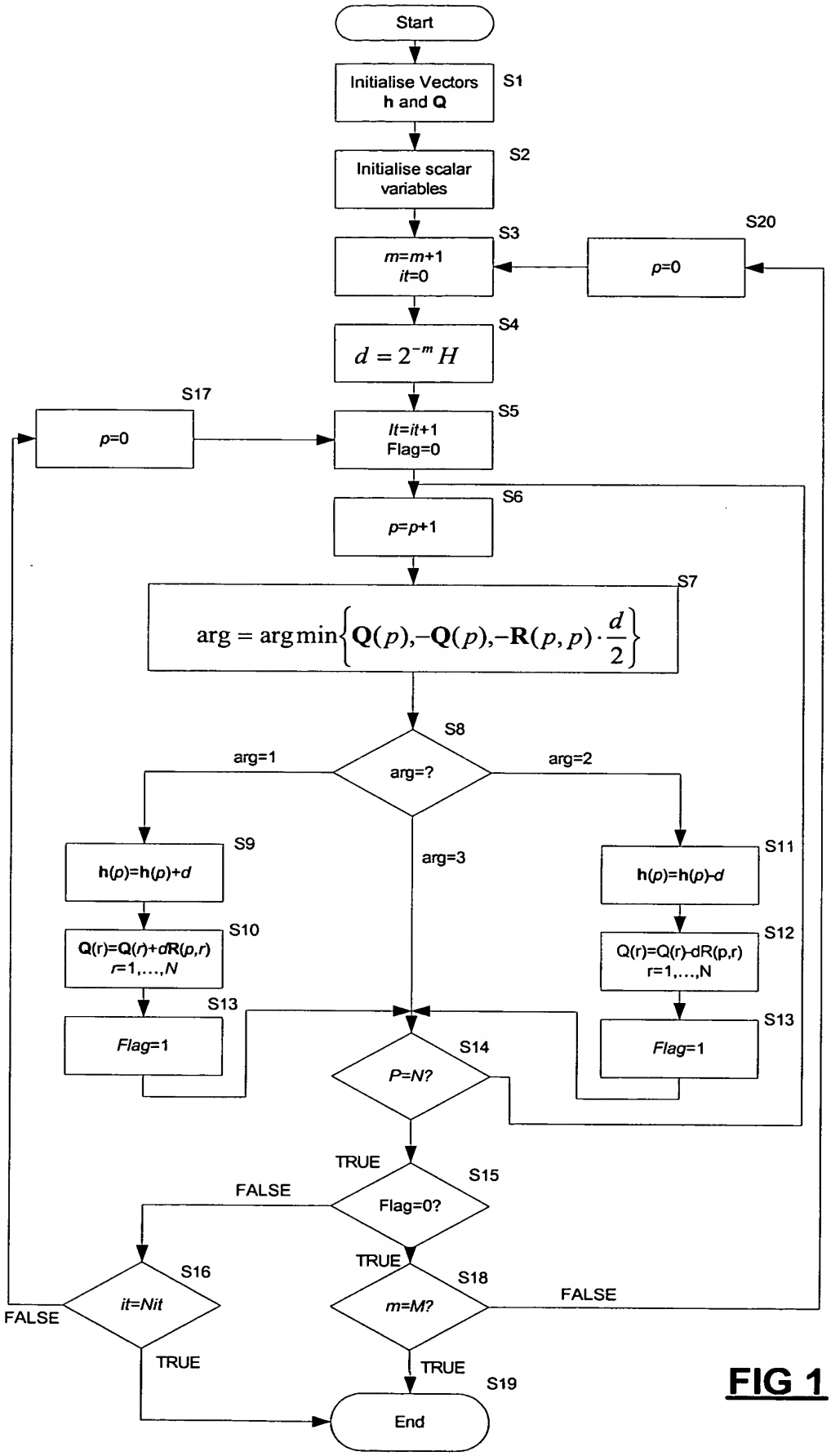
Start
Initialise Vectors h and Q
S1
Initialise scalar variables
S2
m=m+1
it=0
S3
p=0
S20
$d = 2^{-m} H$
S4
p=0
S17
It=it+1
Flag=0
S5
p=p+1
S6
$\arg = \arg\min\left\{\mathbf{Q}(p), -\mathbf{Q}(p), -\mathbf{R}(p,p)\cdot\frac{d}{2}\right\}$
S7
arg=?
S8
arg=1
arg=2
arg=3
h(p)=h(p)+d
S9
h(p)=h(p)-d
S11
Q(r)=Q(r)+dR(p,r) r=1,...,N
S10
Q(r)=Q(r)-dR(p,r) r=1,...,N
S12
Flag=1
S13
Flag=1
S13
P=N?
S14
TRUE
Flag=0?
S15
FALSE
it=Nit
S16
FALSE
TRUE
m=M?
S18
TRUE
FALSE
TRUE
End
S19

FIG 1

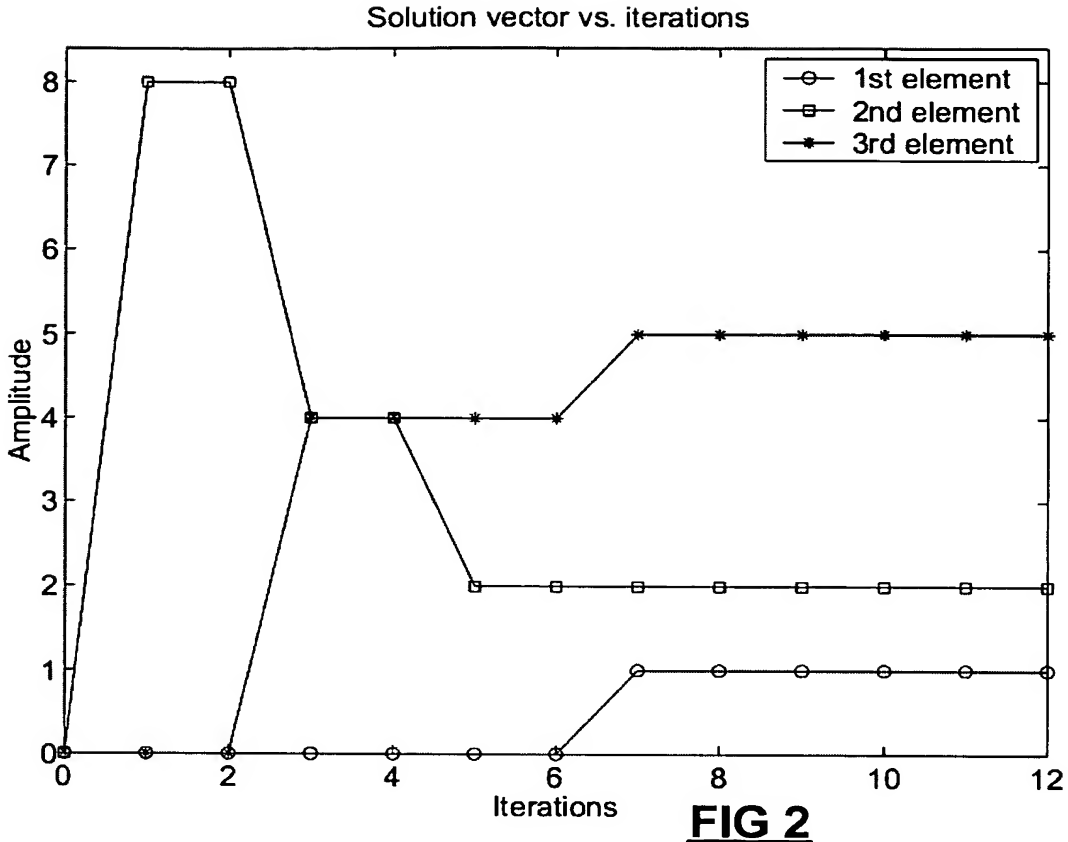
Solution vector vs. iterations
1st element
2nd element
3rd element
Amplitude
Iterations
0
1
2
3
4
5
6
7
8
0
2
4
6
8
10
12

FIG 2

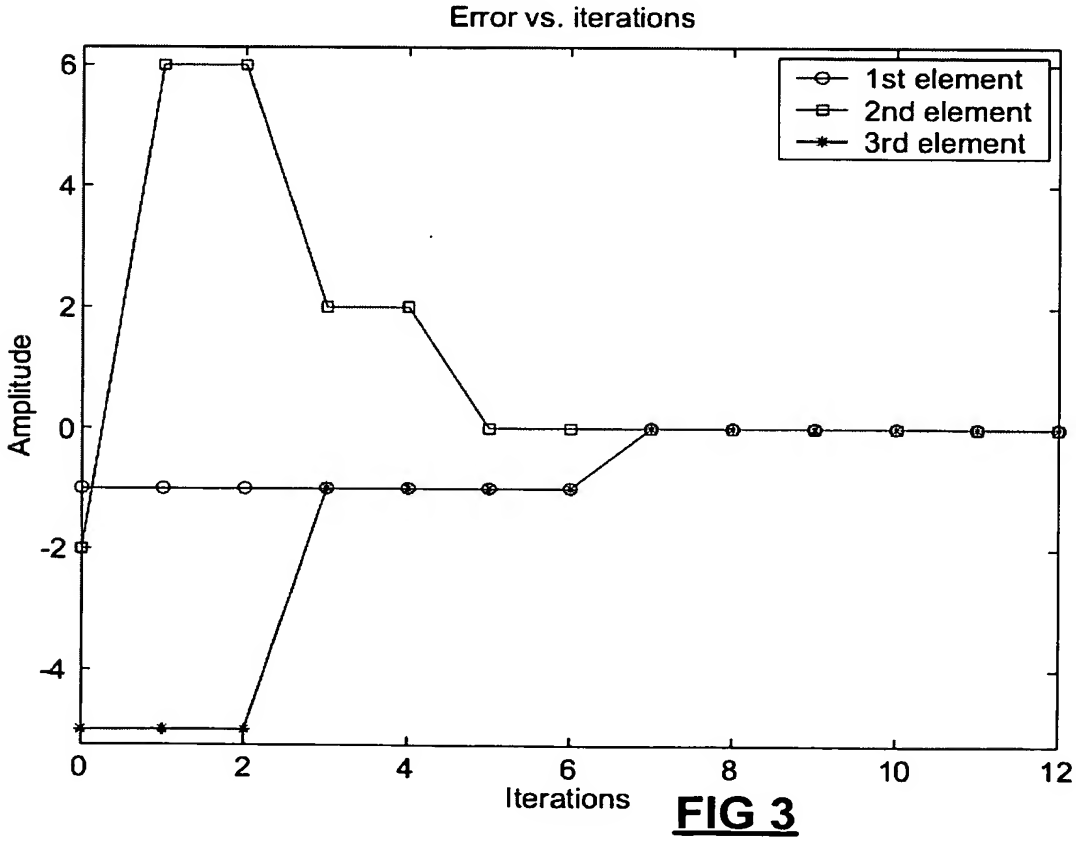
Error vs. iterations
1st element
2nd element
3rd element
Amplitude
Iterations
6
4
2
0
-2
-4
0
2
4
6
8
10
12

FIG 3

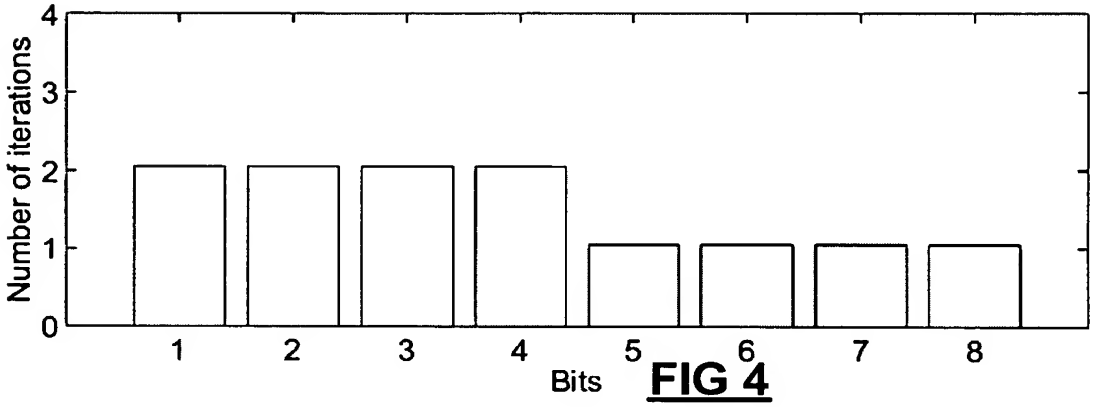
Number of iterations
0
1
2
3
4
1
2
3
4
5
6
7
8
Bits

FIG 4

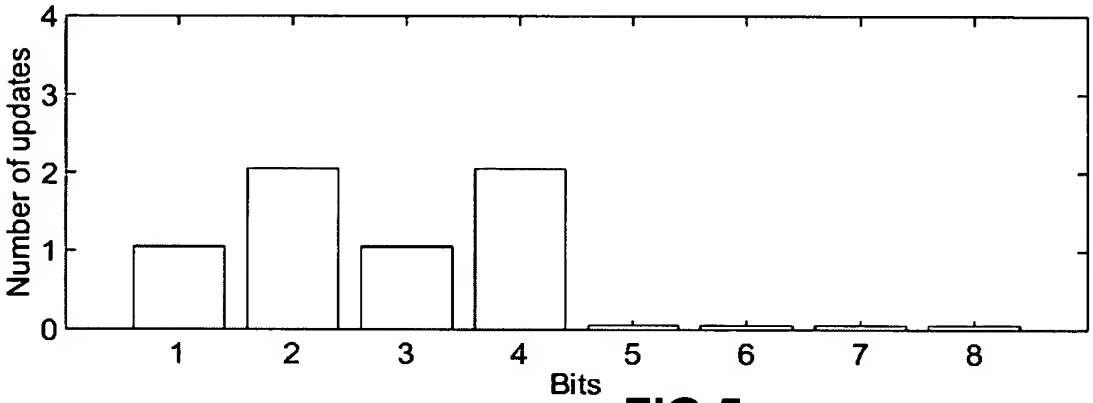
Number of updates
0
1
2
3
4
1
2
3
4
5
6
7
8
Bits

FIG 5

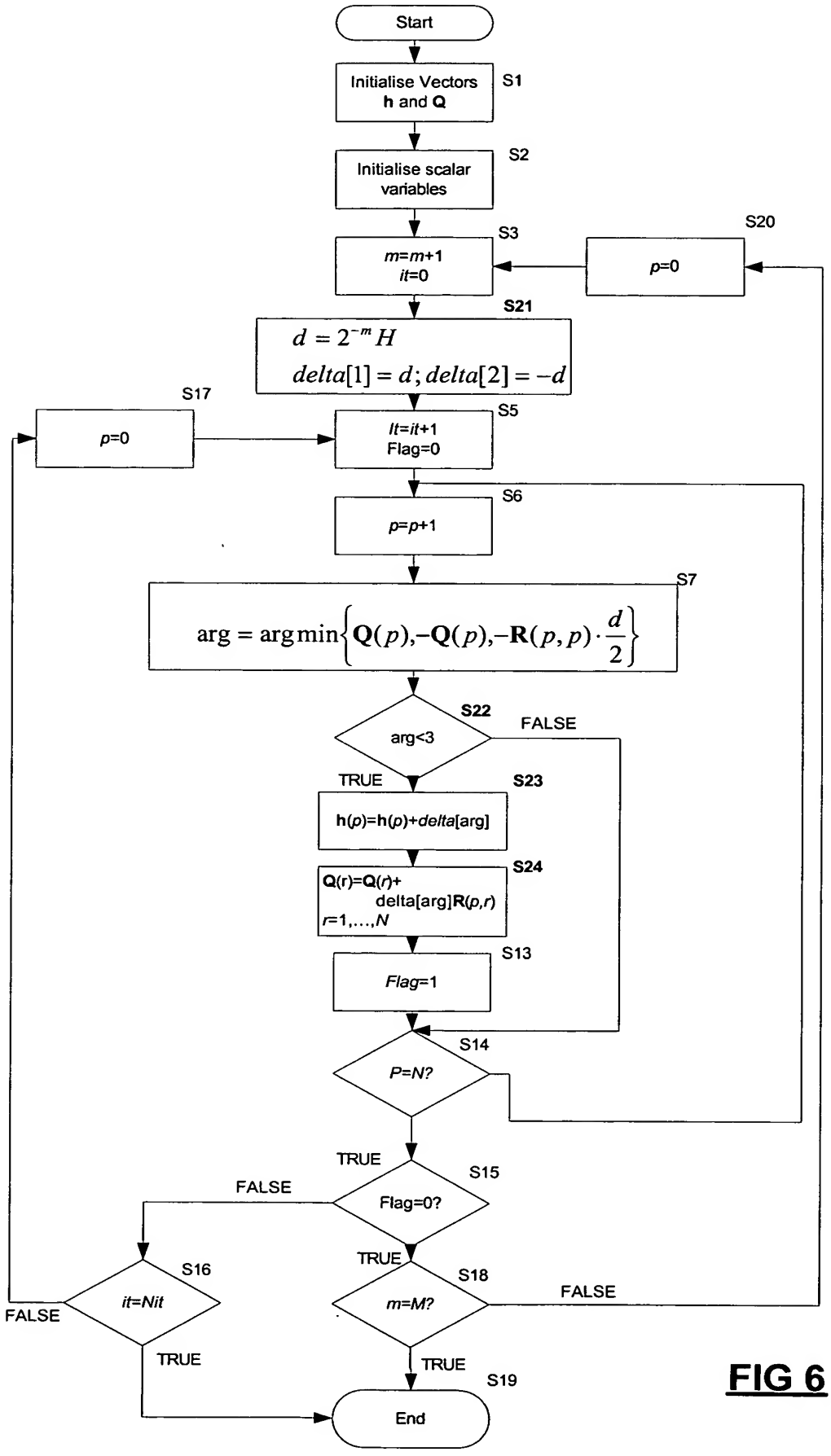
Start
Initialise Vectors h and Q
S1
Initialise scalar variables
S2
m=m+1
it=0
S3
p=0
S20
$d = 2^{-m} H$
$delta[1] = d; delta[2] = -d$
S21
p=0
S17
It=it+1
Flag=0
S5
p=p+1
S6
$\arg = \arg\min\left\{\mathbf{Q}(p), -\mathbf{Q}(p), -\mathbf{R}(p,p)\cdot\frac{d}{2}\right\}$
S7
arg<3
S22
FALSE
TRUE
h(p)=h(p)+delta[arg]
S23
Q(r)=Q(r)+ delta[arg]R(p,r) r=1,...,N
S24
Flag=1
S13
P=N?
S14
TRUE
Flag=0?
S15
FALSE
it=Nit
S16
FALSE
TRUE
m=M?
S18
FALSE
TRUE
End
S19

FIG 6

```
function h=LS_real(R,N,M,H,Nit,beta);
h=zeros(N,1);
delta=zeros(1,2);
Q=-beta;
d=H/2;
for m=1:M
   delta(1)=d; delta(2)=-d;
   d=d/2;
   for it=1:Nit
       Flag=0;
       for p=1:N
           [val arg]=min([Q(p),-Q(p),-R(p,p)*d]);
           if arg<3
              Flag=1;
              h(p)=h(p)+delta(arg);
              Q=Q+delta(arg)*R(p,:);
           end
       end
       if Flag==0 break; end
   end
end

return
```

FIG 7

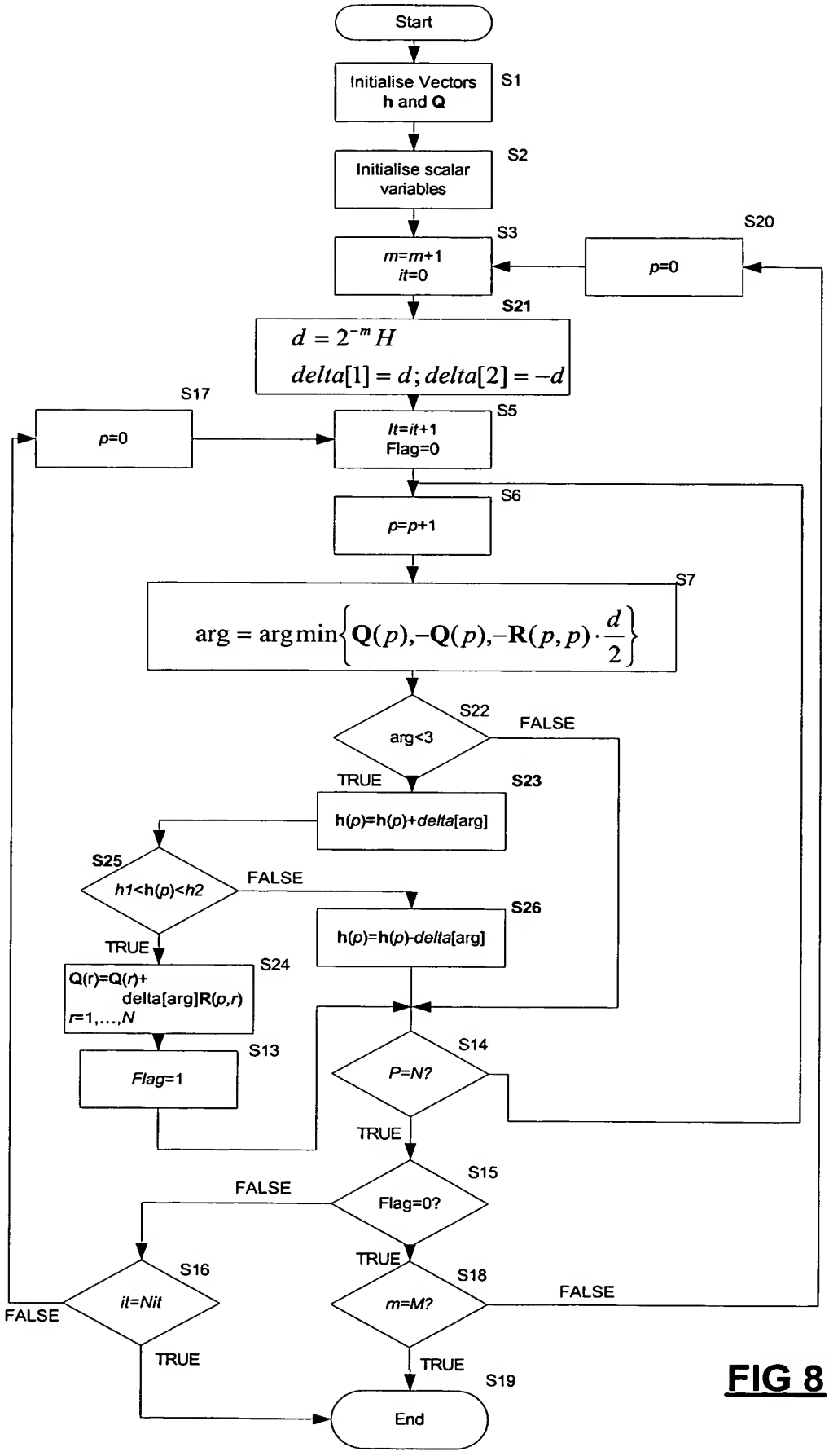
Start
Initialise Vectors h and Q
S1
Initialise scalar variables
S2
m=m+1
it=0
S3
p=0
S20
$d = 2^{-m} H$
$delta[1] = d; delta[2] = -d$
S21
p=0
S17
It=it+1
Flag=0
S5
p=p+1
S6
$\arg = \arg\min\left\{\mathbf{Q}(p), -\mathbf{Q}(p), -\mathbf{R}(p,p)\cdot\frac{d}{2}\right\}$
S7
arg<3
S22
FALSE
TRUE
h(p)=h(p)+delta[arg]
S23
h1<h(p)<h2
S25
FALSE
h(p)=h(p)-delta[arg]
S26
TRUE
Q(r)=Q(r)+ delta[arg]R(p,r) r=1,...,N
S24
Flag=1
S13
P=N?
S14
TRUE
Flag=0?
S15
FALSE
TRUE
it=Nit
S16
FALSE
m=M?
S18
FALSE
TRUE
TRUE
End
S19

FIG 8

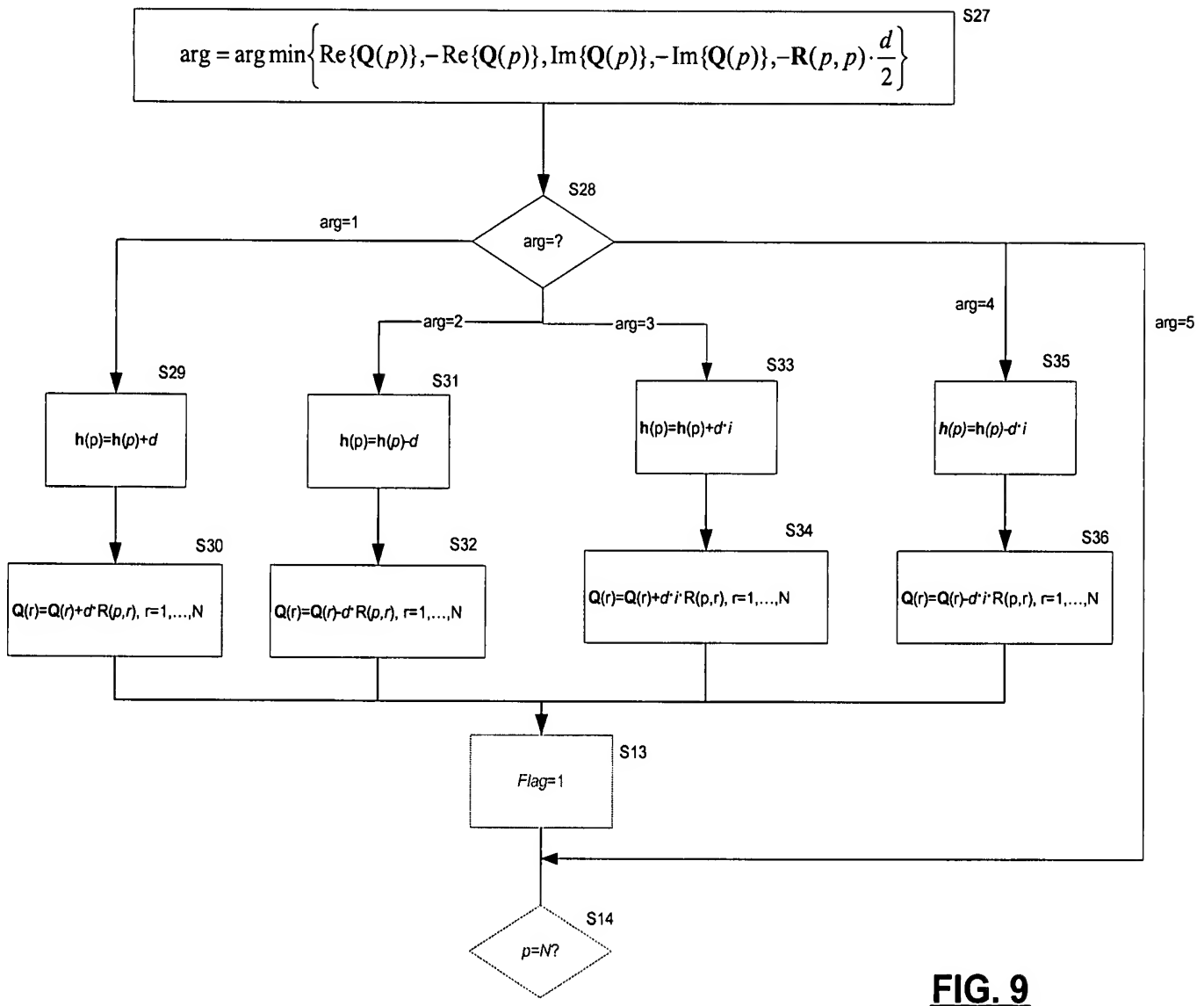
$\arg = \arg\min\left\{\mathrm{Re}\{\mathbf{Q}(p)\}, -\mathrm{Re}\{\mathbf{Q}(p)\}, \mathrm{Im}\{\mathbf{Q}(p)\}, -\mathrm{Im}\{\mathbf{Q}(p)\}, -\mathbf{R}(p,p)\cdot\frac{d}{2}\right\}$
S27
S28
arg=?
arg=1
arg=2
arg=3
arg=4
arg=5
S29
h(p)=h(p)+d
S30
Q(r)=Q(r)+d*R(p,r), r=1,...,N
S31
h(p)=h(p)-d
S32
Q(r)=Q(r)-d*R(p,r), r=1,...,N
S33
h(p)=h(p)+d*i
S34
Q(r)=Q(r)+d*i*R(p,r), r=1,...,N
S35
h(p)=h(p)-d*i
S36
Q(r)=Q(r)-d*i*R(p,r), r=1,...,N
S13
Flag=1
S14
p=N?

FIG. 9

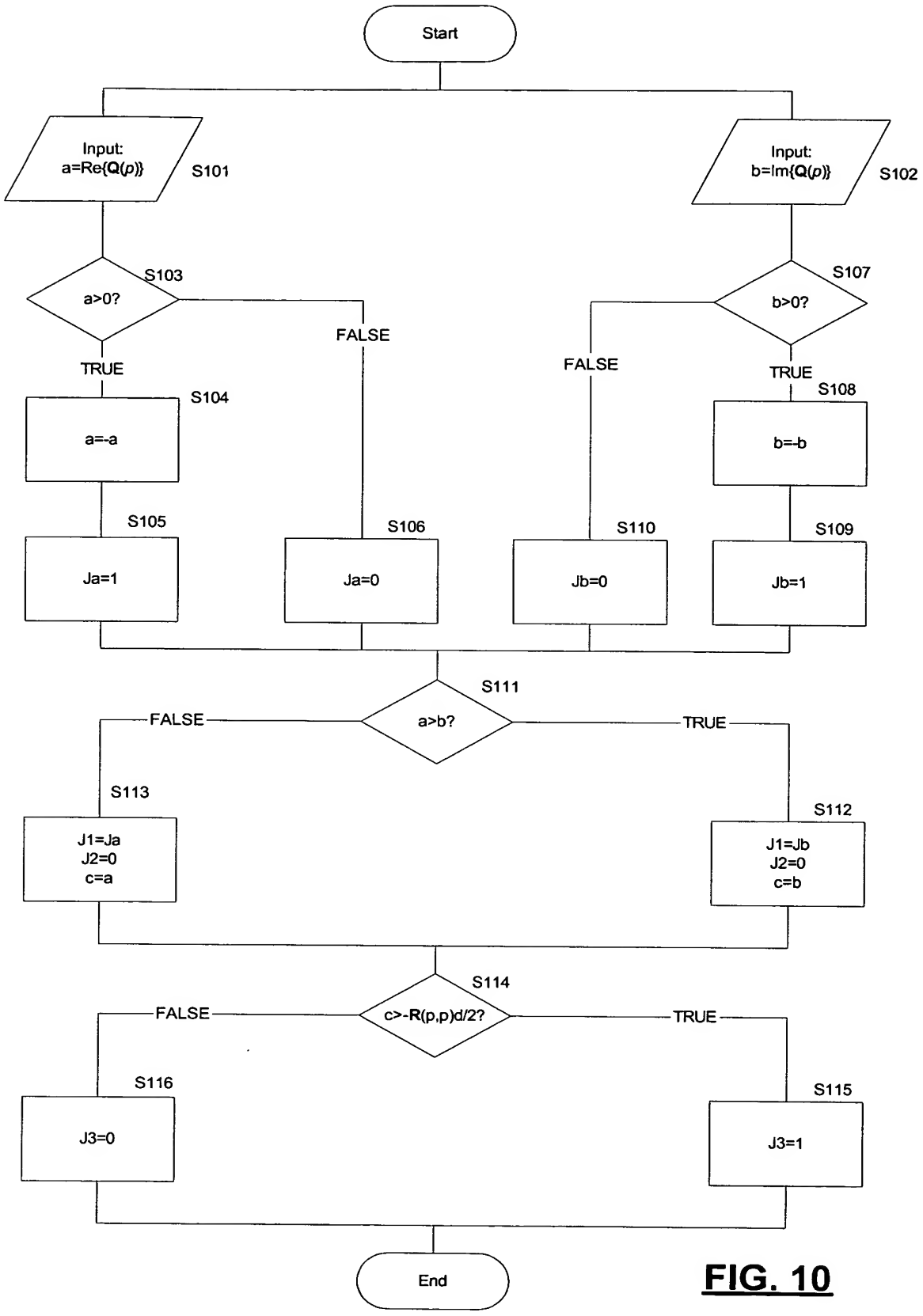
Start
Input:
a=Re{Q(p)}
S101
Input:
b=Im{Q(p)}
S102
S103
a>0?
FALSE
TRUE
S104
a=-a
S105
Ja=1
S106
Ja=0
S107
b>0?
FALSE
TRUE
S108
b=-b
S110
Jb=0
S109
Jb=1
S111
a>b?
FALSE
TRUE
S113
J1=Ja
J2=0
c=a
S112
J1=Jb
J2=0
c=b
S114
c>-R(p,p)d/2?
FALSE
TRUE
S116
J3=0
S115
J3=1
End

FIG. 10

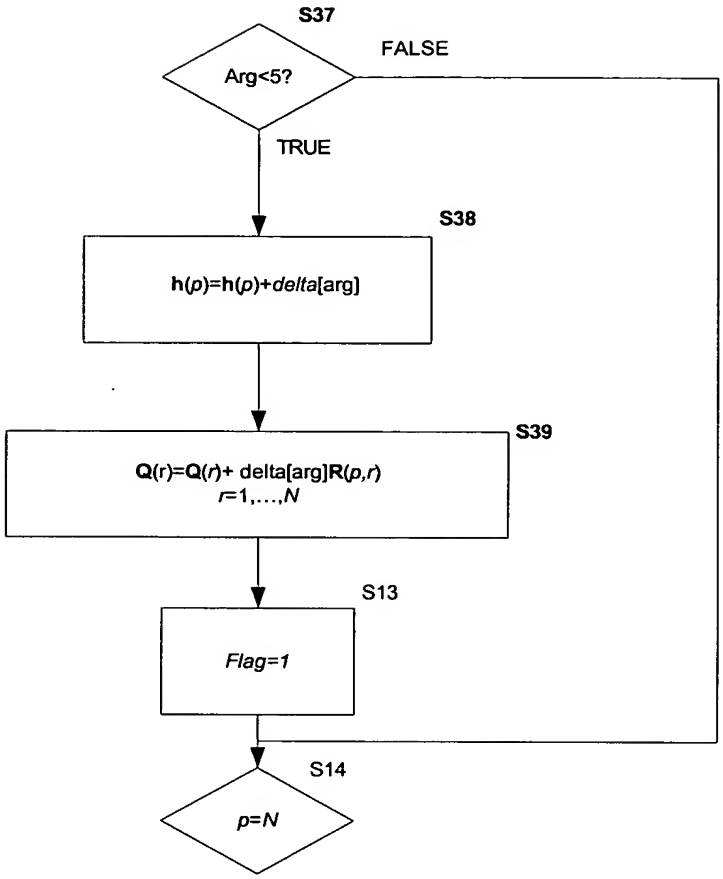
S37
Arg<5?
FALSE
TRUE
S38
h(p)=h(p)+delta[arg]
S39
Q(r)=Q(r)+ delta[arg]R(p,r)
r=1,...,N
S13
Flag=1
S14
p=N

FIG 11

```
function h=LS_cmplx(R,N,M,H,Nit,beta);
h=zeros(N,1);
delta=zeros(1,4);
Q=-beta;
d=H/2;
for m=1:M
   delta(1)=d; delta(2)=-d; delta(3)=i*d; delta(4)=-i*d;
   d=d/2;
   for it=1:Nit
       Flag=0;
       for p=1:N
           [val arg]=min([real(Q(p)),-real(Q(p)),...
                          imag(Q(p)),-imag(Q(p)),-R(p,p)*d]);
           if arg<5
              Flag=1;
              h(p)=h(p)+delta(arg);
              Q=Q+delta(arg)*R(p,:);
           end
       end
       if Flag==0 break; end
   end
end
return
```

FIG 12

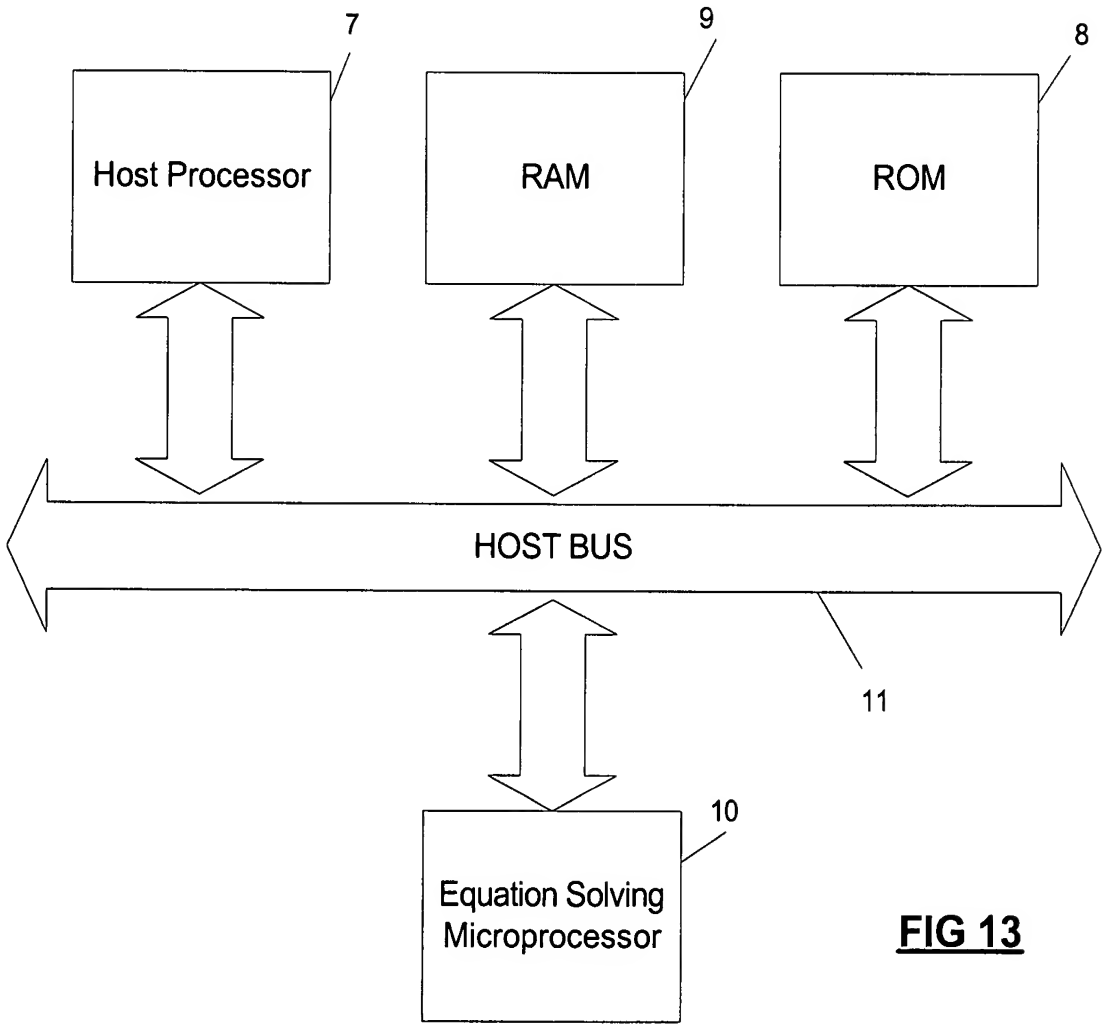
7
9
8
Host Processor
RAM
ROM
HOST BUS
11
10
Equation Solving
Microprocessor

FIG 13

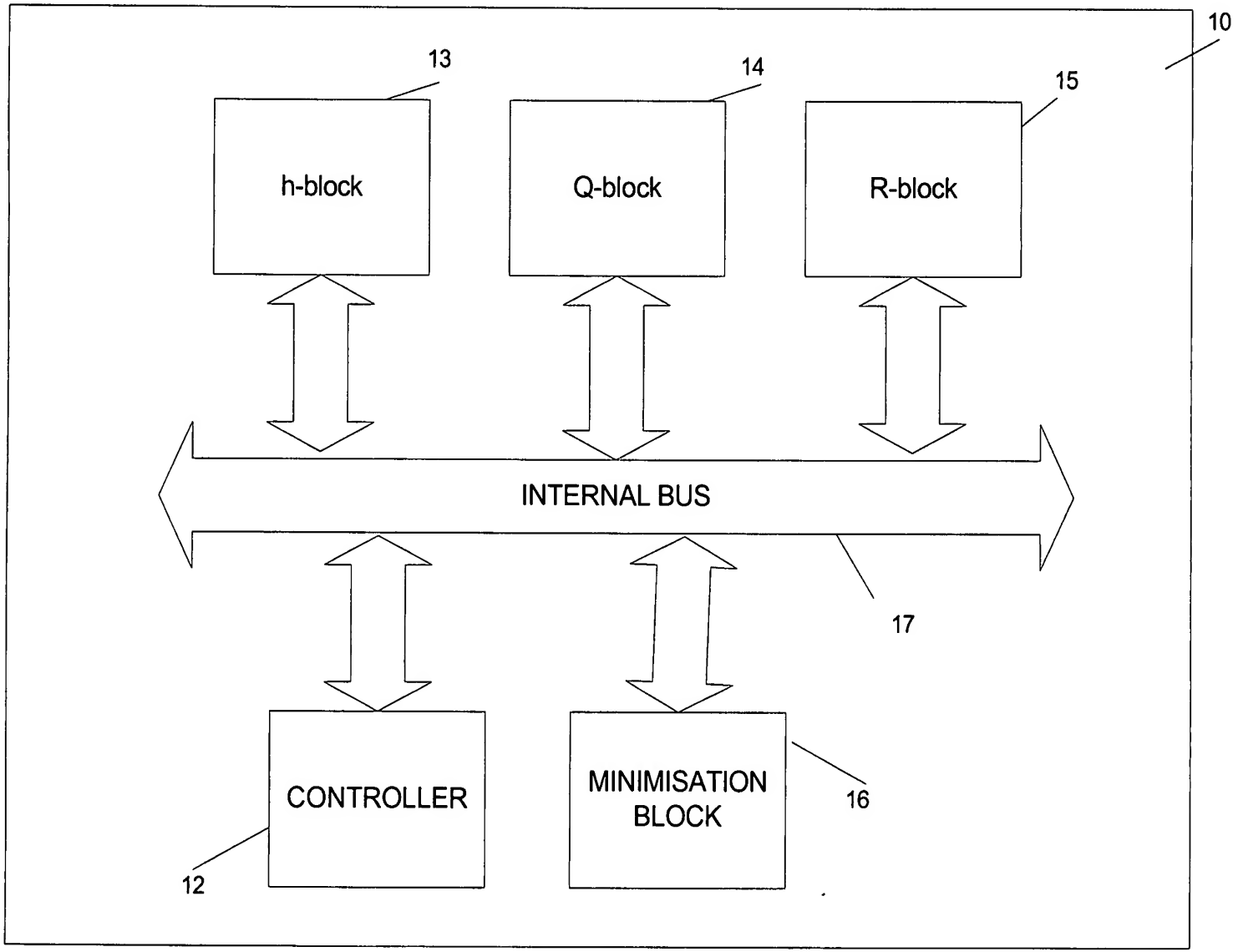
10
13
14
15
h-block
Q-block
R-block
INTERNAL BUS
17
CONTROLLER
MINIMISATION BLOCK
16
12

FIG 14

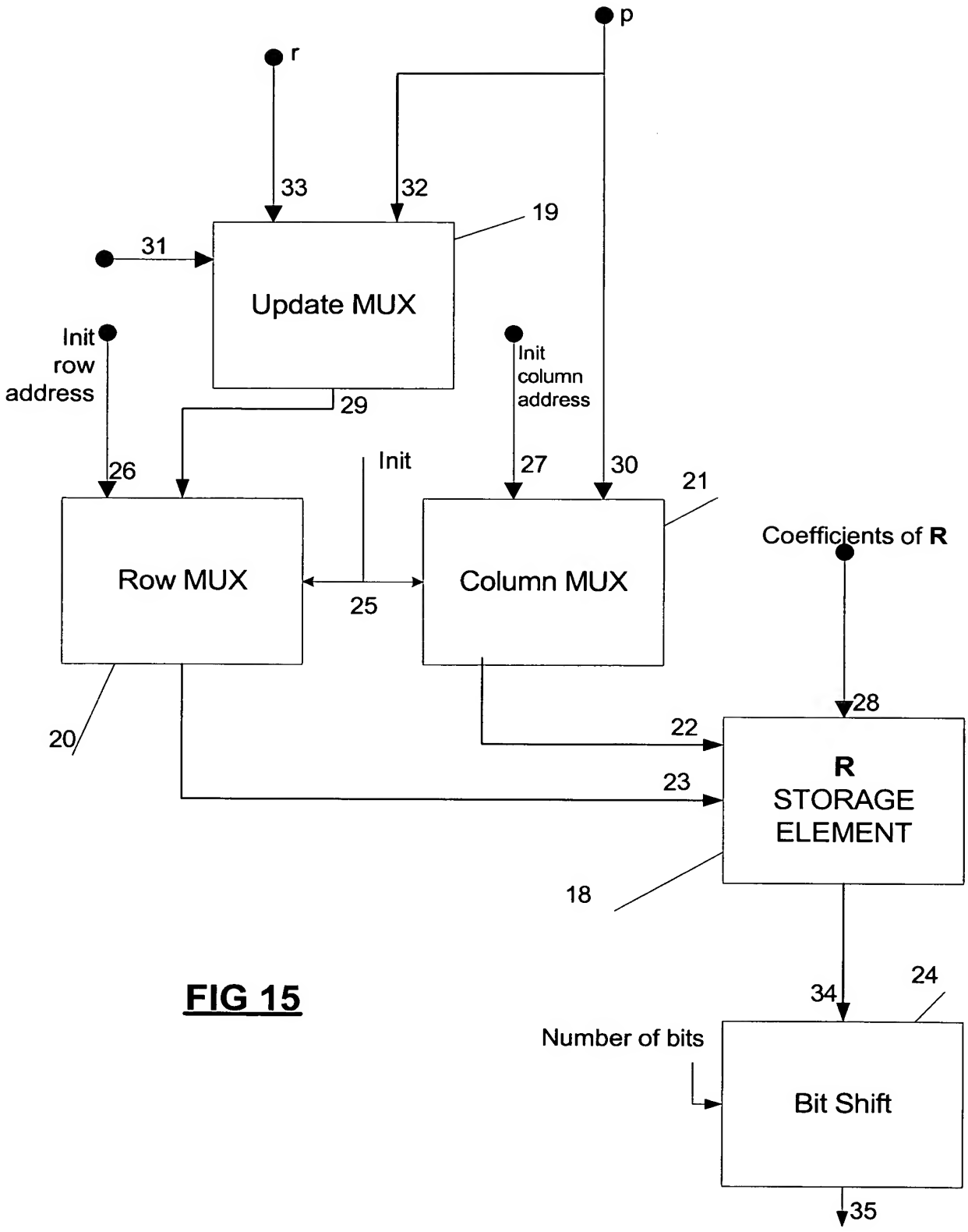
p
r
33
32
19
31
Update MUX
Init
row
address
Init
column
address
29
26
Init
27
30
21
Coefficients of R
Row MUX
25
Column MUX
20
22
28
23
R
STORAGE
ELEMENT
18
24
34
Number of bits
Bit Shift
35

FIG 15

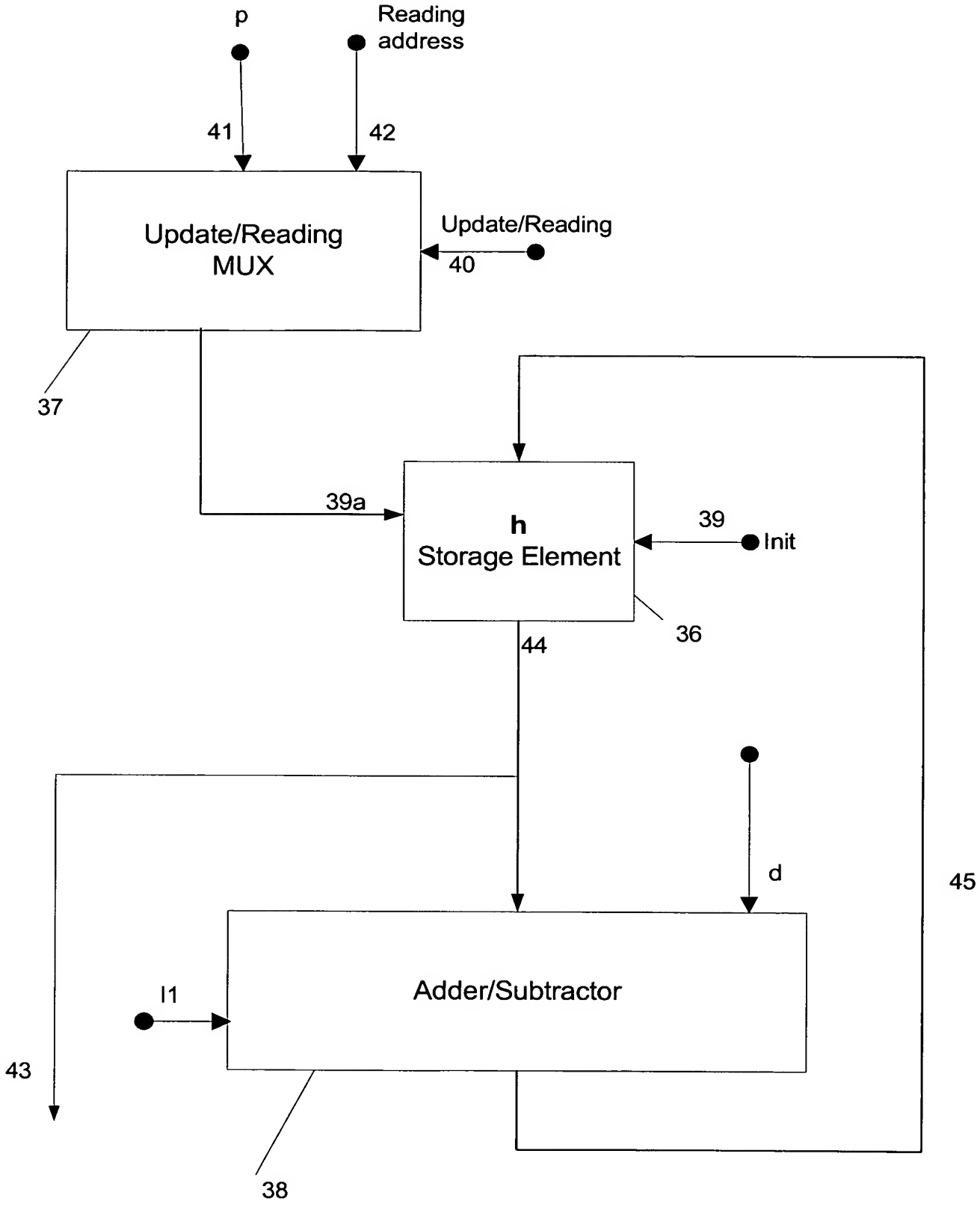
p
Reading address
41
42
Update/Reading MUX
Update/Reading
40
37
39a
h
Storage Element
39
Init
36
44
d
45
Adder/Subtractor
I1
43
38

FIG 16

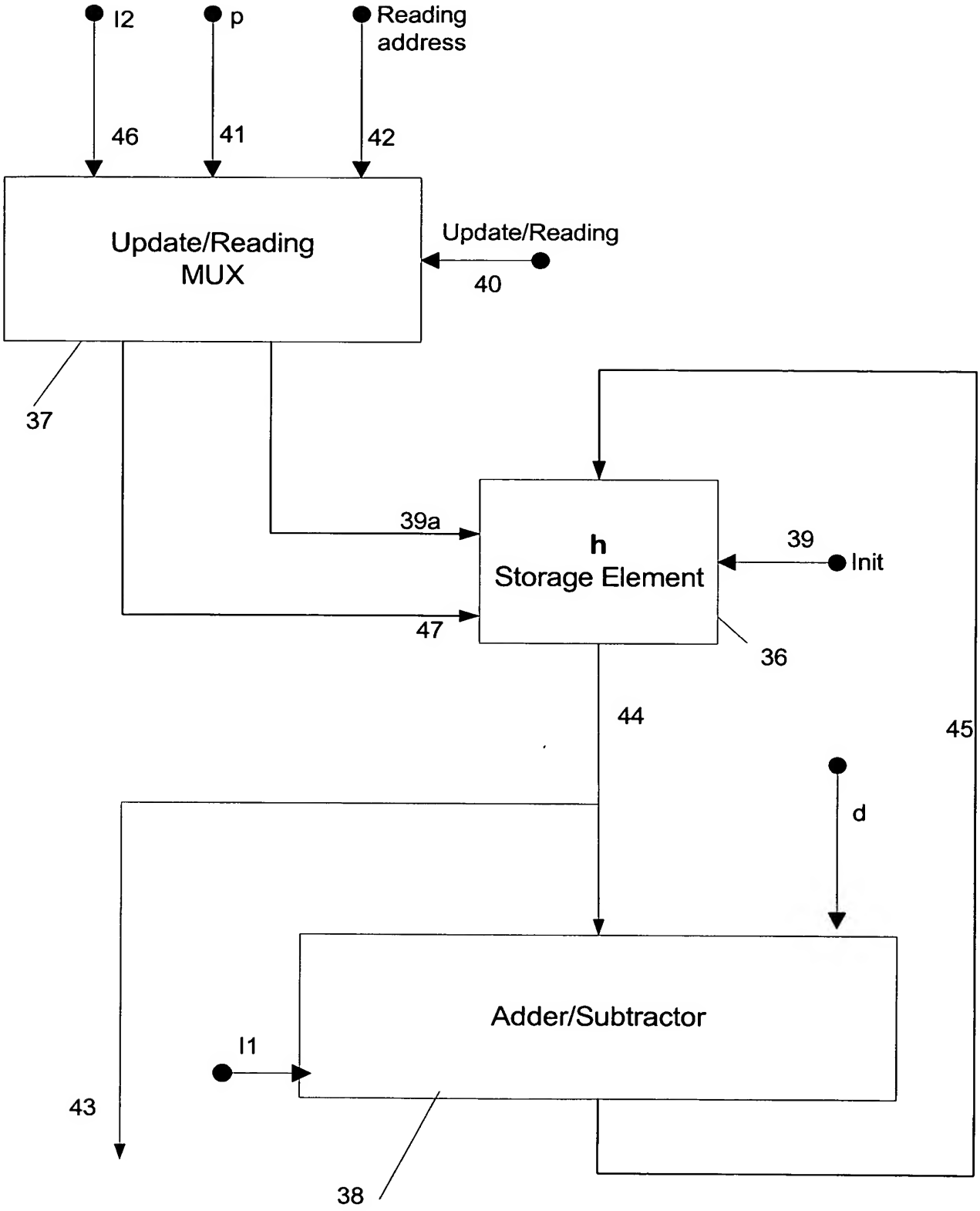
I2
p
Reading address
46
41
42
Update/Reading MUX
Update/Reading
40
37
39a
h
Storage Element
39
Init
47
36
44
45
d
Adder/Subtractor
I1
43
38

FIG 17

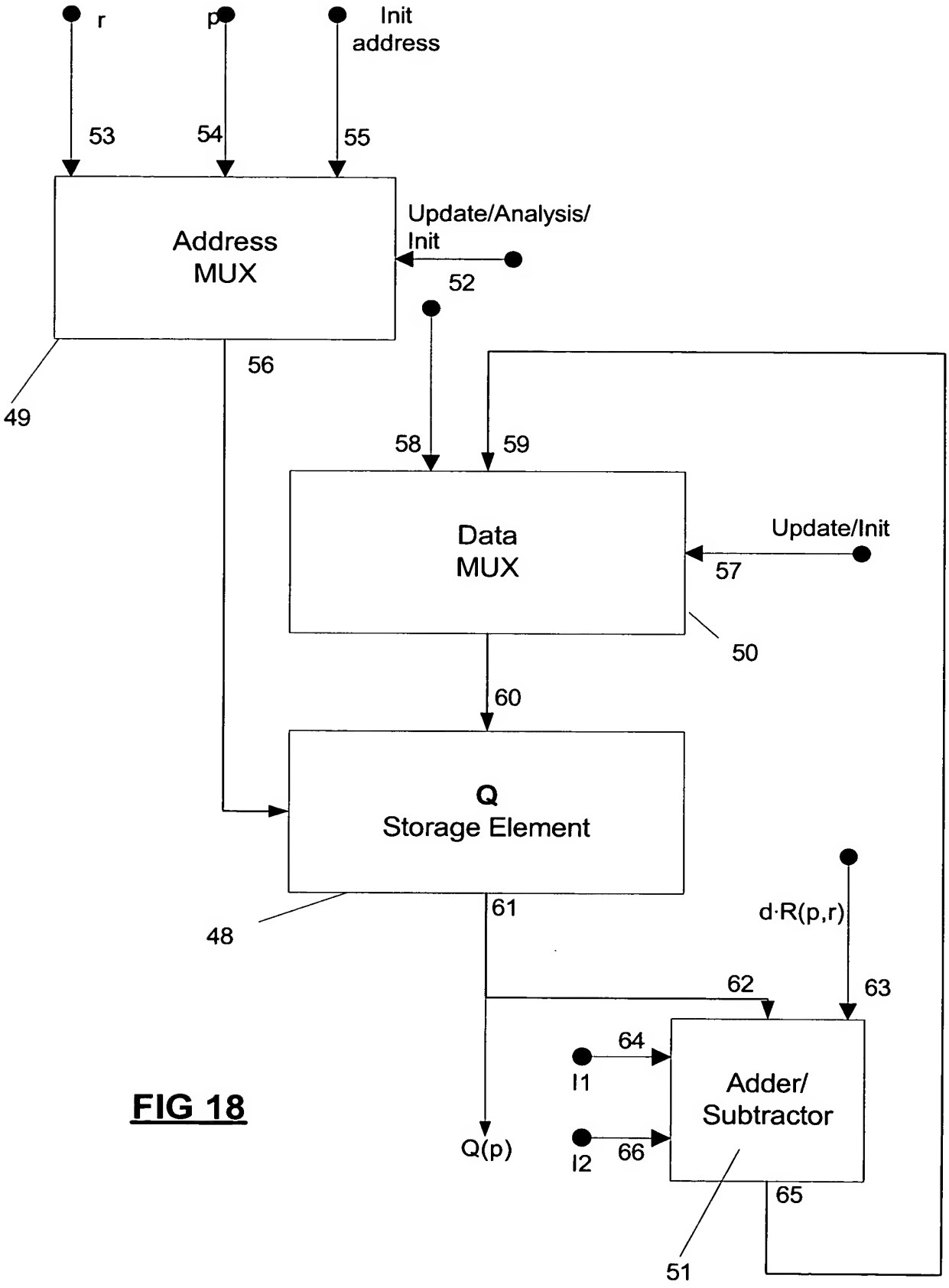
r
p
Init address
53
54
55
Address MUX
Update/Analysis/ Init
52
56
49
58
59
Data MUX
Update/Init
57
50
60
Q Storage Element
48
61
d·R(p,r)
62
63
64
I1
Adder/ Subtractor
Q(p)
I2
66
65
51

FIG 18

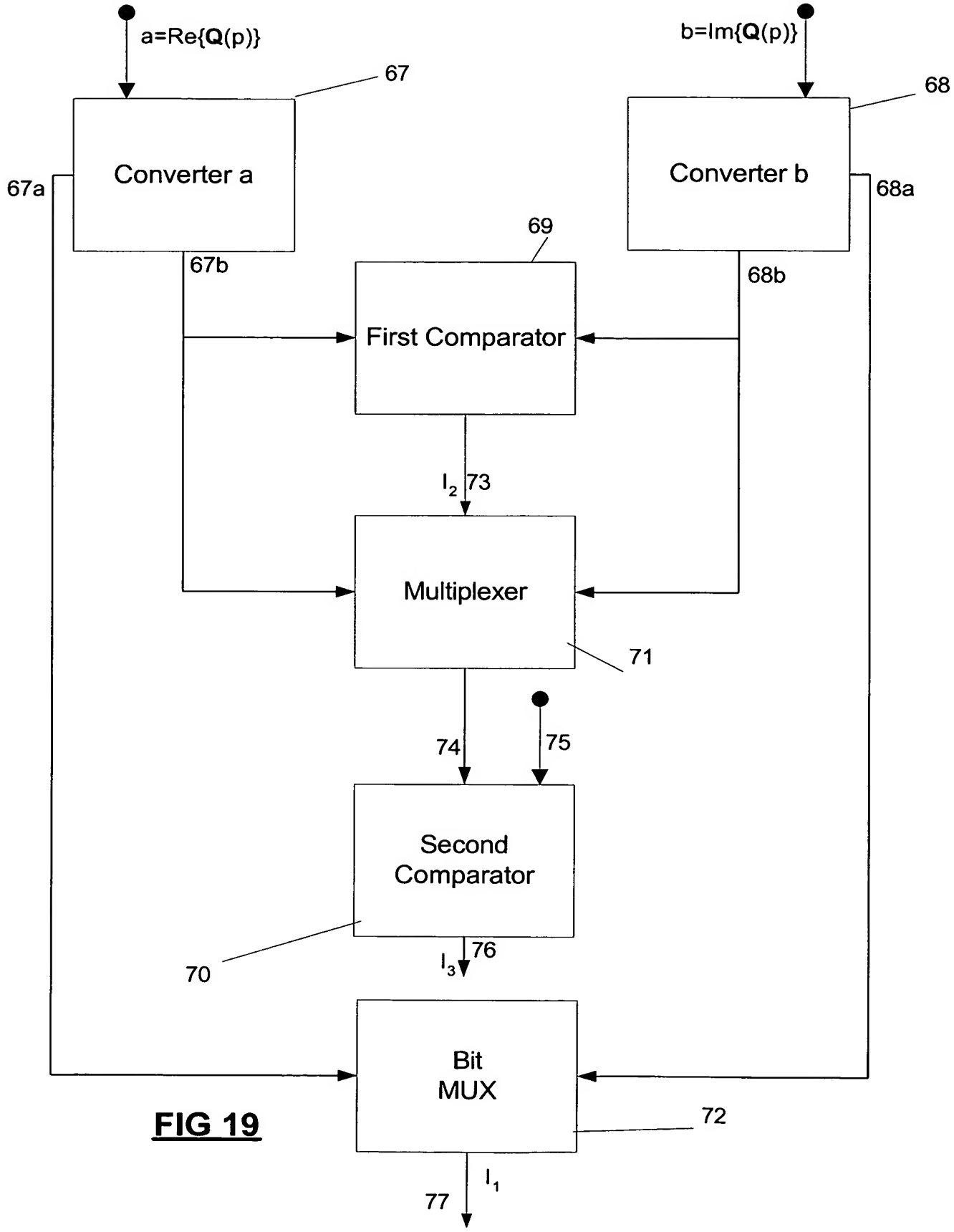
a=Re{Q(p)}
b=Im{Q(p)}
67
68
Converter a
Converter b
67a
68a
69
67b
68b
First Comparator
I_2 73
Multiplexer
71
74
75
Second Comparator
70
I_3 76
Bit MUX
72
I_1
77

FIG 19

```
function h=LS_real_beta(R,N,M,H,Nit,beta);
h=zeros(N,1);
delta=zeros(1,2);
d=H/2;
for m=1:M
   delta(1)=d; delta(2)=-d;
   d=d/2;
   for it=1:Nit
       Flag=0;
       for p=1:N
           [val arg]=min([-beta(p),beta(p),-R(p,p)*d]);
           if arg<3
              Flag=1;
              h(p)=h(p)+delta(arg);
              beta=beta-delta(arg)*R(p,:);
           end
       end
       if Flag==0 break; end
   end
end
return
```

FIG 19a

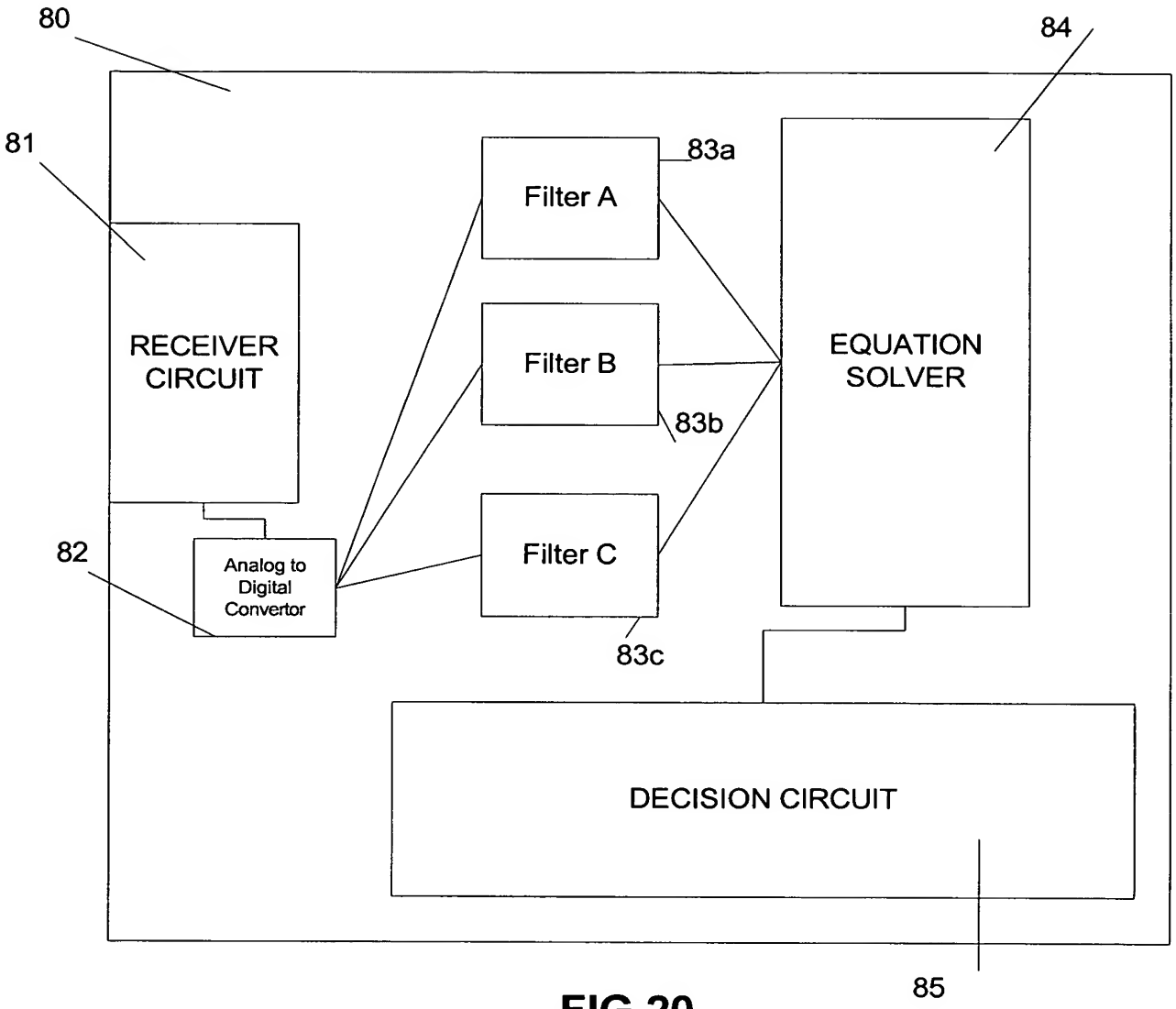
80
84
81
83a
Filter A
RECEIVER CIRCUIT
Filter B
EQUATION SOLVER
83b
82
Analog to Digital Convertor
Filter C
83c
DECISION CIRCUIT
85

FIG 20

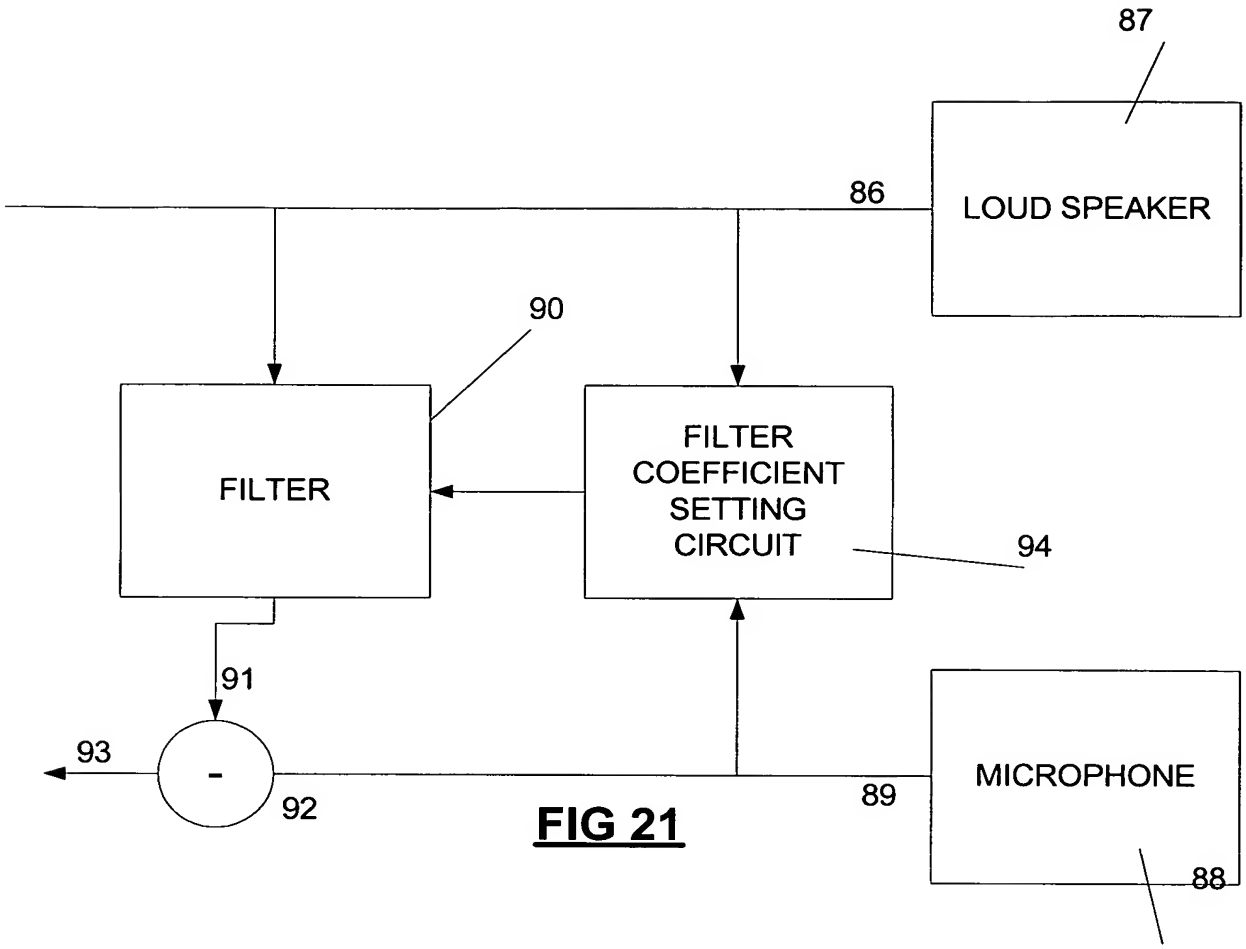
87
LOUD SPEAKER
86
90
FILTER
FILTER COEFFICIENT SETTING CIRCUIT
94
91
93
-
92
89
MICROPHONE
88

FIG 21

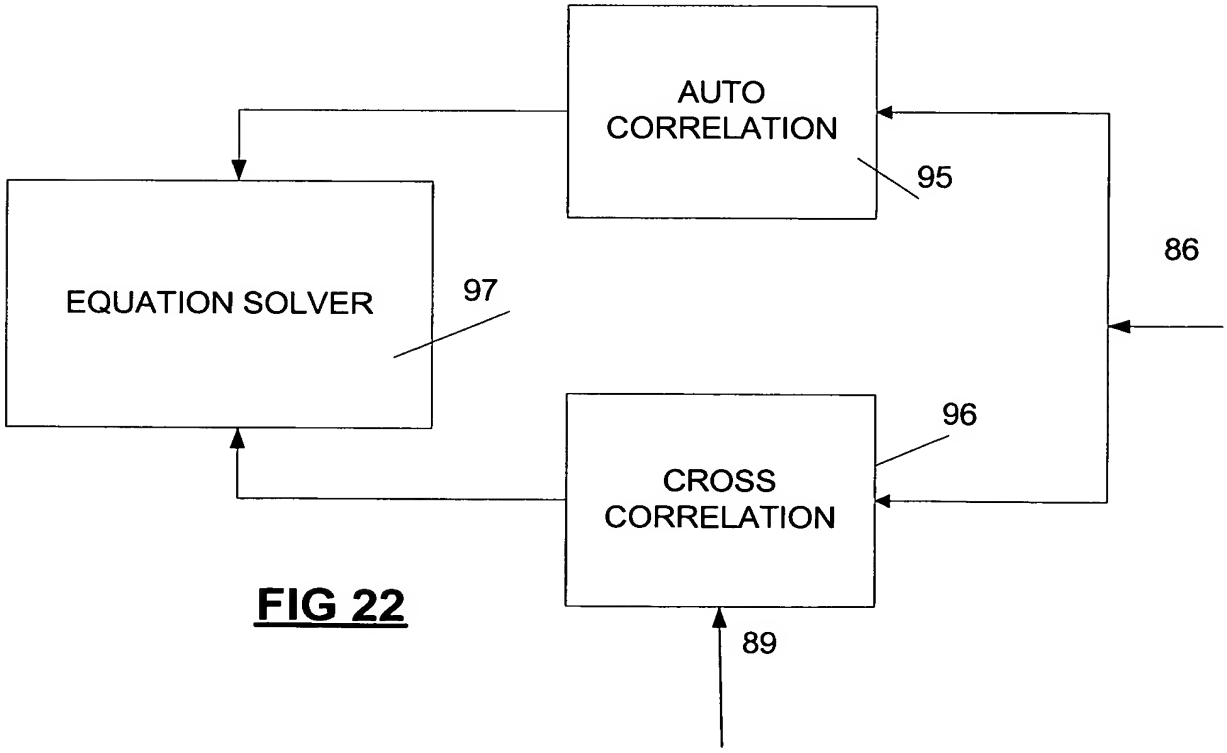
AUTO CORRELATION
95
86
EQUATION SOLVER
97
96
CROSS CORRELATION
89

FIG 22